Total Pages—5

MCA-4

ADA

Set-1

ANALYSIS AND DESIGN OF ALGORITHM

Full Marks : 70

Time : 3 hours

Answer **Q. No. 1** which is compulsory and any **five** from the rest

The figures in the right-hand margin indicate marks

1. Answer *all* questions : 2 × 10

(*a*) What do you mean by divide and conquer approach ? Show the correctness of merge sort algorithm.

(*b*) How many comparisons are necessary in the worst case to find both Maximum and Minimum of '*n*' numbers.

(*c*) Show that after all edges are processed by connected-components, two vertices are in the same connected component if and only if they are in the same set.

(*d*) Use recursion tree method to determine a good asymptotic upper bound on the recurrence $T(n) = 3\,T(\lfloor n/2 \rfloor) + n$.

(*e*) Show the operation of HEAP-EXTRACT-MAX on the heap
$A = \langle 15, 13, 9, 5, 12, 8, 7, 4, 0, 6, 2, 1 \rangle$

(*f*) Show how quicksort performs under the assumption of balanced versus unbalanced partitioning.

(*g*) If $T(n) = 8T(n/2) + n^2$, then find $O(T(n))$.

(*h*) Generate as optimal Huffman code for the following set of frequencies :

$a = 4$ $b = 5$ $c = 7$ $d = 8$ $e = 10$ $f = 12$ $g = 20$

(*i*) Find an optimal solution to the knapsack instance $n = 5$, $m = 45$, $p[1:5] = (18, 20, 30, 17, 24)$ and $w[1:5] = (20, 5, 10, 15, 7)$.

(*j*) What is the smallest value of 'n' such that an algorithm whose running time is $100\,n^2$ runs faster than as algorithm whose running time is 2^n on the same machine ?

2. (*a*) Discuss the advantages and disadvantages of Backtracking technique, giving examples. 5

(*b*) Prove that any comparison sort algorithm requires $\Omega(n \log n)$ comparisons in the worst case. 5

3. (*a*) Show that with the array representation for sorting an n-element heap, the leaves are the nodes indexed by

$$\lfloor n/2 \rfloor + 1, \lfloor n/2 \rfloor + 2, \ldots\ldots, n.$$ 5

(*b*) Illustrate the Bucket Sort algorithm on the array of elements
$A = (0{\cdot}79, 0{\cdot}13, 0{\cdot}16, 0{\cdot}64, 0{\cdot}39, 0{\cdot}20, 0{\cdot}89, 0{\cdot}53, 0{\cdot}71, 0{\cdot}42)$. 5

4. (*a*) Why do we analyze the expected running time of a randonized algorithm and not its worst-case-running time ? 5

(*b*) Consider a hash table of size $m = 1000$ and a hash function, $h(k) = \lfloor m (kA \bmod 1) \rfloor$ for A =(sqrt(5) 1)/2. Compute the locations to which the keys 61, 62, 63, 64 and 65 are mapped. 5

5. (*a*) Classify single source, shortest path algorithms, giving an example in each case. 5

(*b*) Find the minimum number of scalar multiplications and an optimal parenthesization of a matrix-chain product whose sequence of dimensions is given by $< 5, 10, 3, 5, 6 >$. Show the contents of tables '*m*' and '*s*'. 5

6. (*a*) Describe the binary search algorithm and write the pseudo-code for its implementation. 5

(*b*) Differentiate between accounting method and potential method of amortized analysis with suitable examples. 5

7. (*a*) Compare and contrast BFS and DFS search algorithms, discussing their suitability to problem domains.

(*b*) What are the properties of a flow in a flow network ? State the max-flow min-cut theorem. Show that if f_1 and f_2 are flows, then so is $\alpha f_1 + (1 - \alpha) f_2$ for all α in the range $0 \leq \alpha \leq 1$. 5

8. (a) Write an algorithm in dynamic programming that is used for string matching and emulate over an example. 5

(b) Define the class P, NP and NPC. Discuss the circuit-satisfiability problem with an example circuit and prove that it is NP-complete. 5